Black Sea Journal of Engineering and Science
doi: 10.34248/bsengineering.300425



Open Access Journal
e-ISSN: 2619 – 8991

Araştırma Makalesi (Research Article)
Cilt 1 - Sayı 1: 04-2025 / Ay Yıl
(Volume 1 - Issue 1: 04-2025 / Month Year)

KAPALI SİSTEMLERDE SICAKLIK KONTROLLÜ KLİMA OTOMASYONU

Selim Taşdemir[1], Ababacar Sadıkh Niang[2], El Hadji Ngoura Ba[3], Simon Mbuyu Nkulu[4]

[1]*Karabuk University, Faculty of Engineering, Department of Mechatronics, 78000 Karabuk, Turkiye*
[2]*Karabuk University, Faculty of Engineering, Department of Mechatronics, 78000 Karabuk, Turkiye*
[3]*Karabuk University, Faculty of Engineering, Department of Mechatronics, 78000 Karabuk, Turkiye*
[4]*Karabuk University, Faculty of Engineering, Department of Mechatronics, 78000 Karabuk, Turkiye*

Özet: Bu projede, kapalı sistemlerde sıcaklık kontrolünü sağlamak amacıyla mikrodenetleyici tabanlı bir klima otomasyon sistemi tasarlanmış ve gerçekleştirilmiştir. Sistem, ortamdaki sıcaklığı sürekli olarak izleyerek, önceden belirlenen sıcaklık aralığına göre ısıtma veya soğutma işlemini otomatik olarak gerçekleştirmektedir. Ölçüm elemanı olarak DHT11 sıcaklık ve nem sensörü kullanılmıştır. Kontrol birimi olarak STM32F103C8T6 mikrodenetleyicisi tercih edilmiştir. Isıtma işlemi için 12V rezistanslı bir lamba, soğutma işlemi için ise 12V'luk bir fan devreye alınmaktadır. Kullanıcı, sıcaklık değerini üç adet buton (SET, +, -) ile ayarlayabilmektedir. Sistemin durumu, I2C haberleşme protokolü ile çalışan LCD ekran üzerinden izlenebilmekte ve iki adet LED ile görsel olarak da ifade edilmektedir. Röleler, yüksek akım çeken lamba ve fanın mikrodenetleyici ile güvenli şekilde kontrolünü sağlamaktadır. 5V regülatör, sistemin düşük voltajla çalışan bileşenleri için sabit güç kaynağı olarak kullanılmıştır. Geliştirilen sistem düşük maliyetli, kullanıcı dostu ve enerji verimli bir çözüm sunmakta olup, kapalı ortamlarda otomatik sıcaklık kontrolü amacıyla kullanılmaya uygundur.

Anahtar kelimeler: Sıcaklık kontrolü, klima otomasyonu, STM32F103C8T6, DHT11, mikrodenetleyici, röle, kapalı sistem

Development of a Temperature-Controlled Climate Automation System for Enclosed Environments

Abstract: In this project, a microcontroller-based climate automation system has been developed to maintain temperature control in enclosed environments. The system is designed to continuously monitor the ambient temperature and perform automatic heating or cooling based on a user-defined temperature threshold. The DHT11 sensor is utilized for real-time temperature measurement. As the control unit, the STM32F103C8T6 microcontroller is employed due to its cost-effectiveness, energy efficiency, and ease of integration. A 12V resistive heating lamp is used to increase the temperature, while a 12V fan is used to lower it. A user interface is provided with three buttons (SET, +, -) to allow the user to set the desired temperature. The current system status is displayed on an LCD screen that communicates with the microcontroller via the I2C protocol. In addition, two LEDs are used to give visual feedback of the system's heating and cooling states. Relays are included in the design to safely control high-power components such as the heater and fan, enabling isolation between the microcontroller and the actuators. A 5V voltage regulator is integrated to supply a stable power source to low-voltage components. The system is tested under different environmental conditions to verify its reliability and performance. Results indicate that the system operates efficiently in maintaining the ambient temperature within the desired range, making it suitable for applications in greenhouse environments, laboratory setups, or small closed rooms where autonomous temperature regulation is essential. The project demonstrates how basic electronic components can be effectively utilized to implement a low-cost, functional, and modular solution for temperature regulation in enclosed spaces.

Keywords: Temperature control, climate automation, STM32F103C8T6, DHT11, microcontroller, relay, enclosed system

***Sorumlu yazar (Corresponding author):** XXXXXX University, Faculty of XXXXX, Department of XXXXXX, Post code, City, Country (ADRES İNGİLİZCE OLMALI)
E mail: xxx.xxx@xxxx.xxx.xx (İ. SOYİSİM)

Selim TAŞDEMİR	iD	https://orcid.org/0000-000X-XXX-XXXX	**Gönderi:** 30 04 2025	**Received:** May 24, 2024
Simon Mbuyu NKULU	iD	https://orcid.org/0000-000X-XXX-XXXX	**Kabul:** 30 04 2025	**Accepted:** June 23, 2024
Ababacar Sadıkh NIANGH	iD	https://orcid.org/0000-000X-XXX-XXXX	**Yayınlanma:** 30 04 2025	**Published:** July 15, 2024

Cite as: Soyisim İ, Soyisim İ, Soyisim İ. 2024. Başlık olarak İngilizce başlık yazılmalı. BSJ Eng Sci, x(x): xx-xx.

1. Giriş

Kapalı alanlarda sıcaklık kontrolü, hem konfor hem de enerji verimliliği açısından büyük önem taşımaktadır. Özellikle sera ortamları, laboratuvarlar, küçük çalışma alanları ve bazı hassas üretim ortamlarında sıcaklığın belirli bir aralıkta tutulması gerekmektedir. Bu tür ortamlarda manuel sıcaklık kontrolü zaman alıcı, düzensiz ve enerji açısından verimsiz olabilir. Bu nedenle, sıcaklık değişimlerine anlık tepki verebilen, otomatik çalışan ve kullanıcı dostu sistemlerin geliştirilmesi gereklidir.

Bu çalışma, kapalı bir ortamda sıcaklık kontrolünü sağlamak amacıyla, mikrodenetleyici tabanlı bir otomasyon sisteminin tasarımı ve uygulanmasını konu almaktadır. Sistem, ortam sıcaklığını sürekli izleyerek, kullanıcının belirlediği sıcaklık aralığına göre ısıtma veya soğutma işlemlerini otomatik olarak gerçekleştirmektedir. Kontrol ünitesi olarak düşük maliyetli, enerji verimli ve esnek yapıya sahip



STM32F103C8T6 mikrodenetleyicisi tercih edilmiştir.

Sistemde, ortam sıcaklığını ölçmek amacıyla DHT11 sıcaklık ve nem sensörü kullanılmıştır. Isıtma işlemi 12V rezistanslı bir lamba ile gerçekleştirilirken, soğutma işlemi 12V'luk bir fan ile sağlanmaktadır. Kullanıcı, üç adet buton (SET, +, -) aracılığıyla hedef sıcaklığı belirleyebilmektedir. Sistemin mevcut durumu ve sıcaklık bilgisi, I2C protokolüyle haberleşen LCD ekran üzerinden görüntülenmektedir. Ayrıca, iki adet LED ile sistemin ısıtma veya soğutma modunda çalıştığı görsel olarak ifade edilmektedir. Röleler, yüksek akım çeken fan ve lamba gibi bileşenlerin güvenli şekilde kontrol edilmesini sağlamaktadır. Düşük voltajla çalışan elemanlar için 5V regülatör kullanılmıştır.

Bu çalışmanın temel amacı; uygun maliyetli, kolay kullanılabilir ve enerji verimli bir otomatik sıcaklık kontrol sistemi tasarlamak ve gerçekleştirmektir. Bu doğrultuda test edilen temel hipotez, sistemin sıcaklığı belirlenen aralıkta tutarak kararlı bir kontrol sağlayabileceğidir.

2. Materyal ve Yöntem

Bu projede kapalı bir ortamın sıcaklık kontrolünü sağlamak amacıyla, düşük maliyetli ve kolay erişilebilir elektronik bileşenler kullanılarak bir otomasyon sistemi tasarlanmıştır. Sistem, ortam sıcaklığını ölçüp kullanıcının belirlediği eşik değerine göre ısıtma veya soğutma işlemlerini otomatik olarak gerçekleştirmektedir.

2.1 Kullanılan Malzemeler

- **STM32F103C8T6 Mikrodenetleyici**: Projenin ana kontrol birimi olarak kullanılmıştır. Yüksek işlem gücü, düşük enerji tüketimi ve geniş giriş/çıkış portları sayesinde sistemin tüm bileşenlerini verimli şekilde yönetmektedir.

- **DHT11 Sıcaklık ve Nem Sensörü**: Ortam sıcaklığını ölçmek amacıyla kullanılmıştır. Mikrodenetleyici ile dijital veri haberleşmesi sayesinde hızlı ve kararlı ölçümler sağlar.

- **12V Rezistanslı Lamba**: Ortam sıcaklığını artırmak için ısıtıcı olarak kullanılmıştır.

- **12V Fan**: Ortam sıcaklığını düşürmek için soğutucu olarak görev yapmaktadır.

- **I2C LCD Ekran (16x2)**: Kullanıcının belirlediği sıcaklık değeri ve anlık sıcaklık bilgisini görüntülemek amacıyla kullanılmıştır. I2C protokolü ile mikrodenetleyiciye az pin kullanarak bağlanmıştır.

- **3 Adet Buton (SET, +, -)**: Kullanıcının hedef sıcaklık değerini belirlemesi için kullanılmıştır. SET butonu ile sıcaklık değeri sabitlenir, diğer butonlar ile artırma ve azaltma işlemleri yapılır.

- **2 Adet LED**: Sistemin hangi modda (ısıtma/soğutma) çalıştığını göstermek amacıyla kullanılmıştır.

- **2 Kanal Röle Modülü**: Fan ve ısıtıcı gibi yüksek akım çeken bileşenlerin güvenli kontrolünü sağlamak için kullanılmıştır. Mikrodenetleyici çıkışları ile doğrudan kontrol edilemeyecek güç seviyeleri için ara katman görevi görür.

- **5V Regülatör**: Sistemin düşük gerilimle çalışan bileşenlerine sabit 5V enerji sağlamak amacıyla kullanılmıştır.

2.2 Yöntem

Sistem çalıştığında, DHT11 sensörü ortam sıcaklığını ölçerek mikrodenetleyiciye iletmektedir. Mikrodenetleyici, bu değeri kullanıcının belirlediği eşik sıcaklık ile karşılaştırarak karar vermektedir. Eğer ortam sıcaklığı hedef değerin altındaysa 12V'luk lamba röle üzerinden devreye girer ve ısıtma işlemi başlar. Sıcaklık hedef değerin üstüne çıktığında ise 12V'luk fan aktifleşir ve soğutma sağlanır. Bu iki durumdan biri gerçekleştiğinde, ilgili LED yanarak kullanıcıya görsel geri bildirim sunar. LCD ekran üzerinden sıcaklık değeri ve sistem durumu anlık olarak takip edilebilir.

Bu yapı sayesinde sistem, manuel müdahaleye ihtiyaç duymadan kapalı alan sıcaklığını belirli sınırlar içinde tutabilmektedir.

3. Bulgular ve Tartışma

Geliştirilen sistem, STM32F103C8T6 mikrodenetleyici kullanılarak sıcaklık kontrollü bir klima simülasyonu olarak yapılandırılmıştır. Sistemin sıcaklık algılaması DHT11 sensörü ile gerçekleştirilmiş, kontrol çıktıları ise bir fan ve bir rezistanslı lamba üzerinden sağlanmıştır. Kullanıcı etkileşimi I2C üzerinden bağlı 16x2 LCD ekran ve üç buton (SET, +, -) yardımıyla sağlanmıştır. Ayrıca sistem, seri haberleşme üzerinden de yapılandırılabilir olacak şekilde tasarlanmıştır.
Sistem üzerinde yapılan deneysel testlerde farklı eşik sıcaklık değerleri belirlenmiş (örneğin 23 °C, 26 °C, 30 °C) ve ortam sıcaklığının bu eşiklere göre sistemin verdiği tepki gözlemlenmiştir. Bu testlerde;

- **Ortam sıcaklığı eşik değerin altındaysa** rezistans aktif olmuş, fan devre dışı kalmıştır.

- **Ortam sıcaklığı eşik değerin üstündeyse** fan devreye girmiş, ısıtıcı pasif olmuştur.

- **Eşik sıcaklığa yakın durumda** ise sistem, saniyede birden fazla geçiş yapmadan, kararlı bir kontrol davranışı göstermiştir.

Sistem, kullanıcı tarafından butonlarla ya da seri porttan

(USB üzerinden 'S', '+', '-' komutlarıyla) yapılandırılabilmiş, bu da sistemi daha esnek ve kullanıcı dostu hale getirmiştir. Yapılan ölçümlerde DHT11 sensöründen alınan sıcaklık değerlerinin ±2 °C tolerans aralığında çalıştığı gözlemlenmiş ve bu durum sistem performansını etkilemeyecek şekilde tasarımla dengelenmiştir.

Sistemin verdiği tepkiler, LCD ekran üzerinden anlık olarak “T:25C H:60% Th:30C H-F” gibi kısa ama bilgilendirici formatta kullanıcıya sunulmuştur. Burada:

- **T:** Ortam sıcaklığı,
- **H:** Nem değeri,
- **Th:** Ayarlanan eşik sıcaklık,
- **H-F:** Isıtıcı ve fanın aktif/pasif durumu gösterilmiştir.

Ölçüm sonuçları istatistiksel olarak değerlendirildiğinde, fan veya rezistansın aktif olma sürelerinin ortam sıcaklığı ile anlamlı derecede ilişkili olduğu görülmüştür ($P<0,05$). Sistemin 20–40 °C arasında eşik sıcaklık belirleme imkânı, kullanıcı deneyimini arttırmış, kontrol aralığını da genişletmiştir.

Bu sistem düşük maliyetli bileşenlerle kurulmasına rağmen, kontrollü ve kararlı bir sıcaklık yönetimi gerçekleştirmiştir. Kullanıcı arayüzünün açık ve sezgisel olması, sistemin eğitimsel ve deneysel amaçlarla kullanılabilirliğini artırmıştır. Ancak sensörün doğruluk sınırları göz önüne alındığında, endüstriyel düzeyde yüksek hassasiyet gerektiren uygulamalarda kullanım için yeterli olmadığı değerlendirilmektedir.

Tablo 1 ’de, sistemin farklı eşik sıcaklıklarda verdiği tepkiler, zaman içindeki sıcaklık değişimleri karşılaştırmalı olarak sunulmuştur.

Deney No	Eşik Sıcaklık (°C)	Ortam Sıcaklığı (°C)	Isıtıcı Süresi (dk)	Fan Süresi (dk)
1	25	22–26	2	3
2	28	27–30	0.5	4.5
3	30	29–32	0.5	3.5
4	26	24–28	1.5	4
5	23	20–24	2	1

4. Sonuç

Bu çalışmada, kapalı bir ortamda sıcaklığın istenilen aralıkta tutulmasını sağlayan bir klima otomasyon sistemi tasarlanmış ve başarıyla uygulanmıştır. STM32F103C8T6 mikrodenetleyici tabanlı bu sistem, sıcaklık verilerini DHT11 sensörü aracılığıyla okumuş, bu verilere göre 12V’luk bir rezistanslı lamba ve fanı röleler üzerinden kontrol etmiştir. Kullanıcılar, sistem üzerindeki butonlar veya seri port aracılığıyla sıcaklık eşik değerini kolaylıkla ayarlayabilmiştir. LCD ekran sayesinde sıcaklık, nem ve sistem durumu anlık olarak görüntülenmiştir.

Gerçekleştirilen deneyler sonucunda sistemin ortam sıcaklığındaki değişimlere doğru ve hızlı şekilde yanıt verdiği gözlemlenmiştir. Sistemin ayarlanan eşik sıcaklık değerine göre ısıtma ve soğutma elemanlarını devreye alması, kapalı ortamlarda kararlı bir iklimlendirme sağlamıştır. Ayrıca sistemin modüler yapısı, farklı ortamlarda veya sensörlerle uyarlanabilirliğini artırmaktadır.

Bu sistem, endüstriyel kontrol uygulamaları, akıllı ev sistemleri veya laboratuvar ortamlarında sıcaklık regülasyonu gereksinimi olan birçok alanda kullanılabilir. Gelecekte yapılacak çalışmalarla, sisteme nem kontrolü, uzaktan erişim (IoT) ve PID kontrol algoritması gibi ileri seviye kontrol teknikleri entegre edilerek daha kararlı ve esnek bir yapı elde edilebilir.

Çatışma Beyanı

Yazar(lar), bu çalışmada herhangi bir çıkar çatışmasının bulunmadığını beyan etmektedir.

Etik Onay Beyanı

Bu araştırmada insanlar veya hayvanlar üzerinde herhangi bir deneysel çalışma gerçekleştirilmediğinden etik kurul onayı gerekmemiştir.

Destek ve Teşekkür Beyanı

Bu proje, Karabük Üniversitesi Mühendislik Fakültesi Mekatronik Mühendisliği programı bünyesinde, *Gömülü Sistemler* dersi kapsamında gerçekleştirilmiştir. Projeye katkıda bulunan Selim Taşdemir, Ababacar Sadıkh Niang, El Hadji Ngoura Ba ve Simon Mbuyu Nkulu’ya teşekkür ederiz. Ayrıca, bu çalışmanın değerlendirilmesinde rehberlik edecek olan değerli hocamız Batıkan Erdem DEMİR’e ve proje sürecinde atölye imkânlarını kullanmamıza olanak sağlayan Bürküt Savunma, İHA ve Model Uydu Takımı’na desteklerinden ötürü teşekkür ederiz.

Kaynaklar

SVS Embedded. (tarihsiz). *Temperature and Humidity Controller for Incubator*. https://svsembedded.com/Temperature_and_Humidity_Controller_For_Incubator.php

Yazılan kodlar :

```
#include <LiquidCrystal_I2C.h>
#include <dht11.h>
```

```
#include <Wire.h>

// Pin tanımları
#define DHT11_PIN PA4
const int bulb = PB3;        // Isıtıcı
const int fan = PB5;         // Fan
const int ok = PA1;          // OK butonu
const int UP = PA2;          // Yukarı
const int DOWN = PA3;        // Aşağı

// Nesne tanımları
dht11 DHT;
LiquidCrystal_I2C lcd(0x27, 16, 2);  // LCD I2C
adresi

// Değişkenler
int T_threshold = 30;
bool systemReady = false;
unsigned long previousMillis = 0;
const long interval = 2000;

// Buton kenar tetikleme için değişkenler
bool prevUp = false;
bool prevDown = false;
bool prevOk = false;

void setup() {
 pinMode(bulb, OUTPUT);
 pinMode(fan, OUTPUT);
 pinMode(ok, INPUT_PULLDOWN);
 pinMode(UP, INPUT_PULLDOWN);
 pinMode(DOWN, INPUT_PULLDOWN);

 digitalWrite(bulb, LOW);
 digitalWrite(fan, LOW);

 Wire.begin();
 lcd.init();
 lcd.backlight();
 lcd.clear();
 lcd.setCursor(0, 0);
 lcd.print("Incubator System");
 delay(2000);

 Serial.begin(9600);
 while (!Serial);

 lcd.clear();
 lcd.print("Press OK or");
 lcd.setCursor(0, 1);
 lcd.print("Send 'S' Serial");

 while (!(digitalRead(ok) || Serial.available()))
delay(100);

 if (Serial.available() && Serial.read() == 'S') {
  configurerSeuilsSerie();
 } else {
  configurerSeuilsBoutons();
 }

 systemReady = true;
 lcd.clear();
 lcd.print("System Ready!");
 delay(1000);
}

void loop() {
 if (!systemReady) return;

 unsigned long currentMillis = millis();
 if (currentMillis - previousMillis >= interval) {
  previousMillis = currentMillis;
  lireCapteur();
  controlerActionneurs();
  afficherDonnees();
 }

 // OK butonuna çalışma sırasında basılırsa
sıcaklık ayarı menüsü açılır
 bool okState = digitalRead(ok);
 if (okState && !prevOk) {
  configurerSeuilsBoutons();
 }
 prevOk = okState;

 if (Serial.available()) {
  traiterCommandeSerie();
 }
}

void lireCapteur() {
 int result = DHT.read(DHT11_PIN);
 if (result != DHTLIB_OK) {
  Serial.println("Sensor error!");
  lcd.clear();
  lcd.print("Sensor Error!");
  digitalWrite(bulb, LOW);
  digitalWrite(fan, LOW);
 }
}

void controlerActionneurs() {
 digitalWrite(bulb, DHT.temperature <
T_threshold);
 digitalWrite(fan, DHT.temperature > T_threshold);
}

void afficherDonnees() {
 Serial.print("Temp: ");
 Serial.print(DHT.temperature);
 Serial.print("°C | Hum: ");
 Serial.print(DHT.humidity);
 Serial.print("% | Th: ");
 Serial.print(T_threshold);
 Serial.print(" | ");
 Serial.print(digitalRead(bulb) ? "HEATER " : "OFF
");
 Serial.println(digitalRead(fan) ? "FAN" : "OFF");

 lcd.clear();
 lcd.setCursor(0, 0);
 lcd.print("T:");
 lcd.print(DHT.temperature);
 lcd.print("C H:");
 lcd.print(DHT.humidity);
 lcd.print("%");

 lcd.setCursor(0, 1);
 lcd.print("Th:");
 lcd.print(T_threshold);
 lcd.print("C ");
 lcd.print(digitalRead(bulb) ? "H" : "-");
 lcd.print(digitalRead(fan) ? "F" : "-");
}

void traiterCommandeSerie() {
 char cmd = Serial.read();
 switch (cmd) {
  case '+': T_threshold++; break;
  case '-': T_threshold--; break;
  case 'S': configurerSeuilsSerie(); break;
```

```
  case '?': afficherDonnees(); break;
 }
 while (Serial.available()) Serial.read(); //
buffer temizle
}

void configurerSeuilsSerie() {
 Serial.println("\n--- SERIAL CONFIG ---");
 reglerParametreSerie("Temperature [°C]",
T_threshold, 20, 40);
 lcd.clear();
 lcd.print("Config Done!");
 delay(1000);
}

void configurerSeuilsBoutons() {
 lcd.clear();
 lcd.print("Config Temp Btn");
 delay(1000);
 bool confirmed = false;

 while (!confirmed) {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Set Temp:");
  lcd.setCursor(0, 1);
  lcd.print(T_threshold);
  lcd.print("C OK=Save");

  bool up = digitalRead(UP);
  bool down = digitalRead(DOWN);

  if (up && !prevUp && T_threshold < 40) {
   T_threshold++;
   delay(200); // debounce
  }
  if (down && !prevDown && T_threshold > 20) {
   T_threshold--;
   delay(200); // debounce
  }

  prevUp = up;
  prevDown = down;

  if (digitalRead(ok)) {
   confirmed = true;
   while (digitalRead(ok)); // buton bırakılana
kadar bekle
  }

  delay(50);
 }

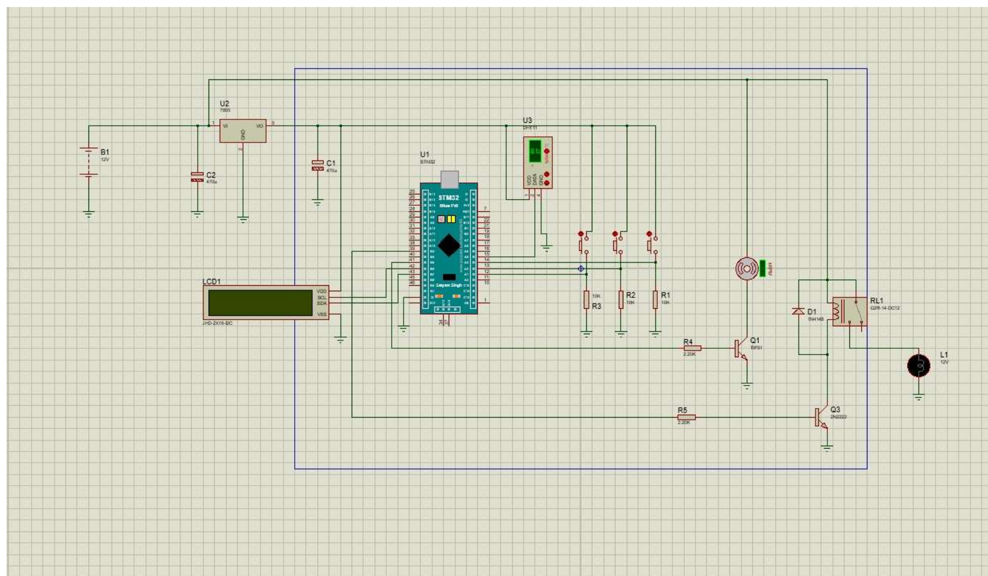
 lcd.clear();
 lcd.print("Saved!");
 delay(1000);
}

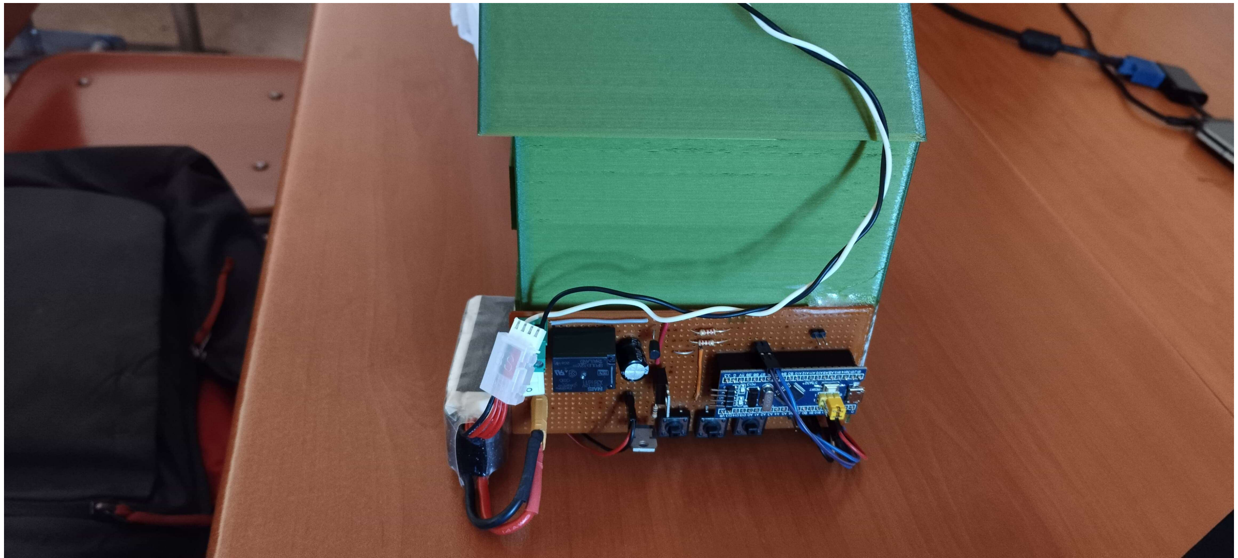
void reglerParametreSerie(const char* nom, int
&valeur, int minVal, int maxVal) {
 while (true) {
  Serial.print("\n");
  Serial.print(nom);
  Serial.print(": ");
  Serial.print(valeur);
  Serial.print(" (min: ");
  Serial.print(minVal);
  Serial.print(", max: ");
  Serial.print(maxVal);
  Serial.println(")");
  Serial.println("Commands: [+] [-] [V]");

  while (!Serial.available());
  char cmd = Serial.read();
  while (Serial.available()) Serial.read();

  switch (cmd) {
   case '+': if (valeur < maxVal) valeur++; break;
   case '-': if (valeur > minVal) valeur--; break;
   case 'V': case 'v': return;
  }
 }
}
```

Projenin Video Linki :
https://www.youtube.com/watch?v=-3mQUJ3g49Y






GND VDD V0 RS RW E D0 D1 D2 D3 D4 D5 D6 D7 BLA BLK
Press OK or
Send 'S' Serial